DATABASE MANAGEMENT SYSTEMS

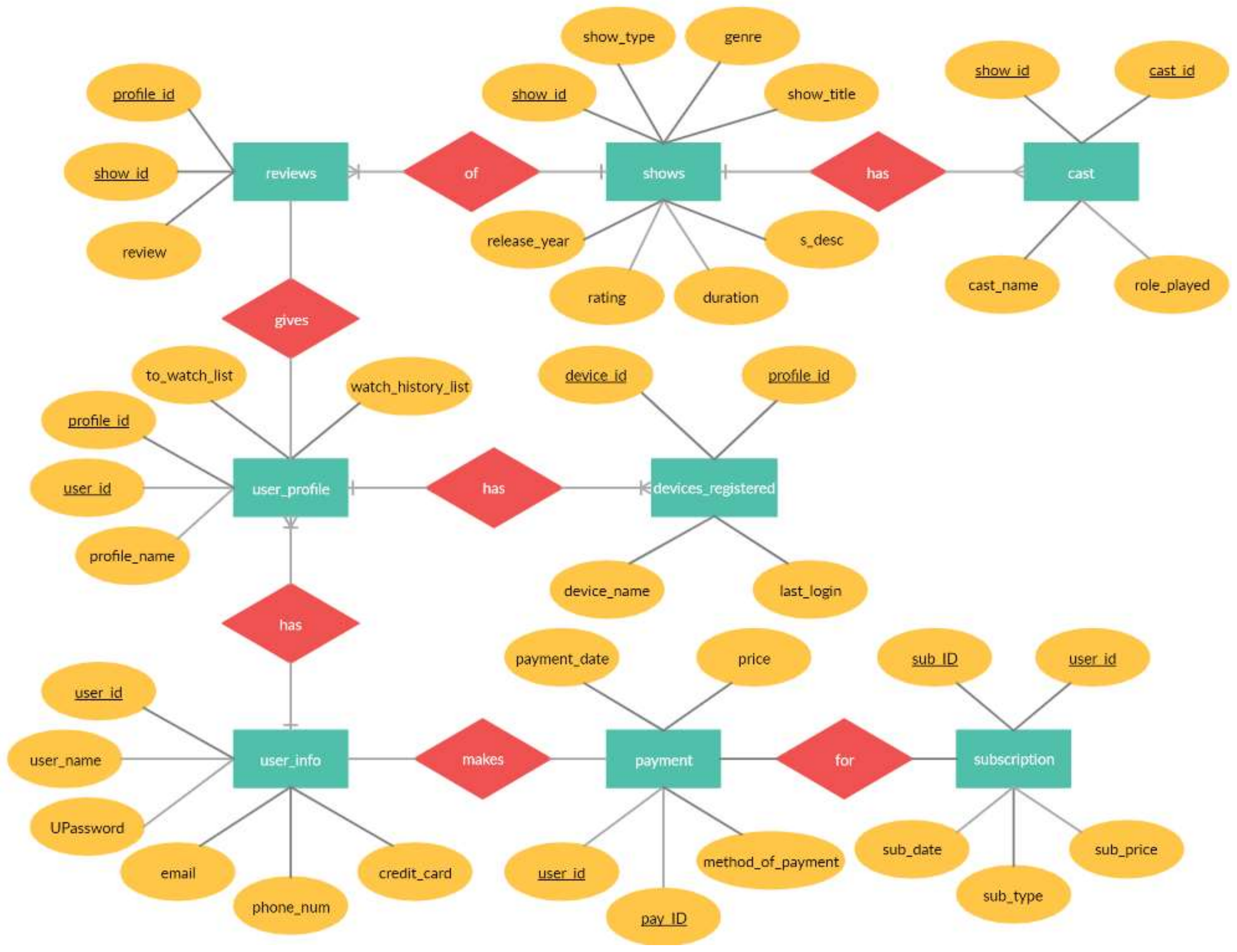
ONLINE STREAMING PLATFORM

REQUIREMENTS

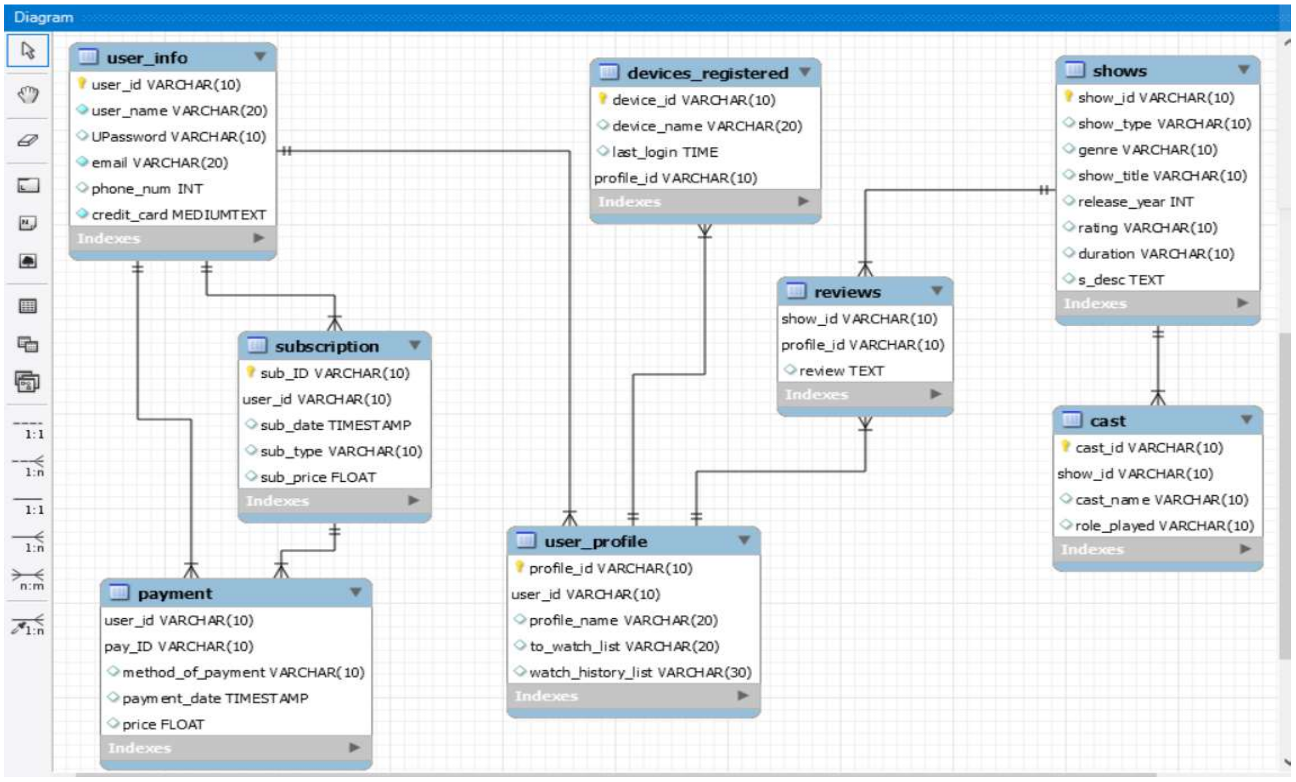
Create a Database Management System for an Online Streaming Platform that satisfies the following requirements:

1. Users should first create their own accounts (which will have user_id, user_name,UPassword, email, phone_num and credit_card)
2. Each account will have a choice between 5 subscription plans and make payment for the same accordingly each month.
3. Each account can then have up to four Profiles for their family members. Each profile will have its individual "Watch History"(what movies were watched on which date), "Watch Later" - the movies to watch in the future), maximum of two devices registered(which will show the last login) and Reviews left on each show.
4. We need to keep track of a show's ShowID, ShowTitle, ShowType(Movie, Documentary), Release Year, Duration, Description and Rating.
5. Since shows are starred by some actors, we need to keep track of the actors and their role in the movie.

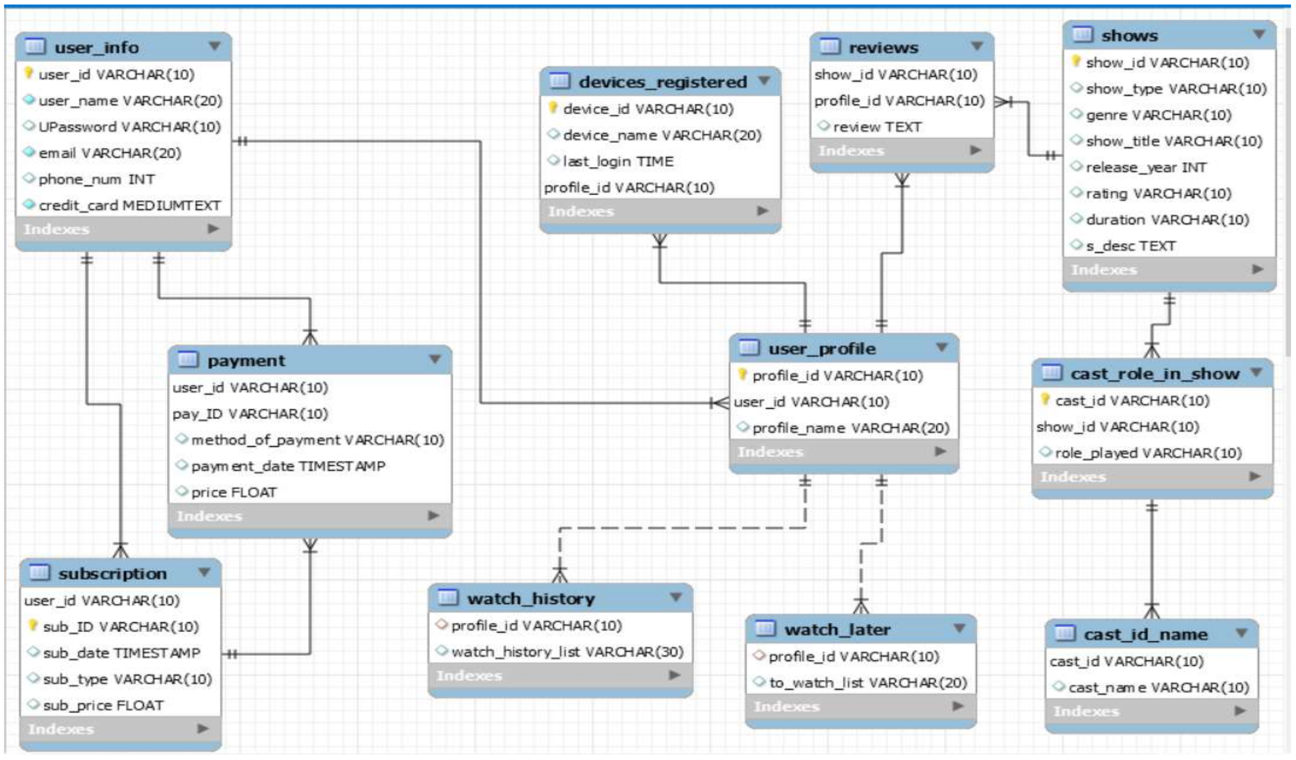
ER DIAGRAM



RELATIONAL MODEL



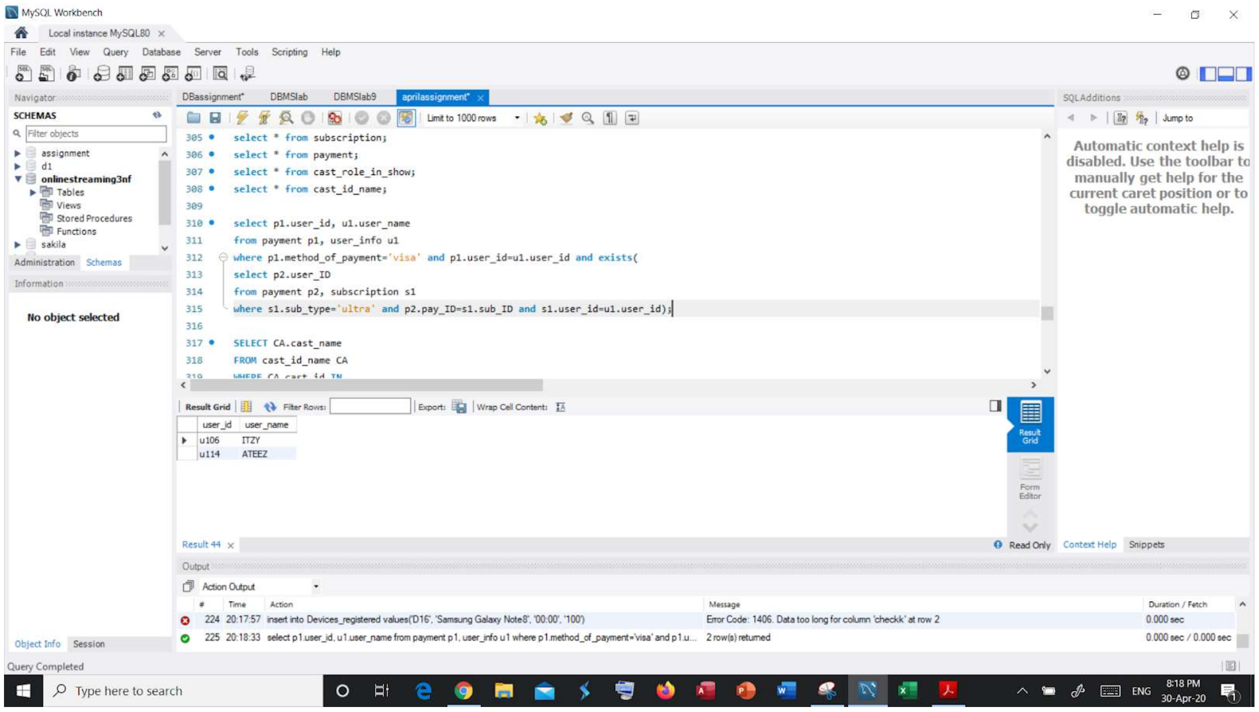
3NF



QUERIES:

Q1. Write a **nested query** to display the IDs and names of the users who paid via Visa and subscribed to Ultra plan.

SELECT p1.user_id, u1.user_name
FROM payment p1, user_info u1
WHERE p1.method_of_payment='visa' AND p1.user_id=u1.user_id AND EXISTS(
SELECT p2.user_ID
FROM payment p2, subscription s1
WHERE s1.sub_type='ultra' AND p2.pay_ID=s1.sub_ID AND s1.user_id=u1.user_id);



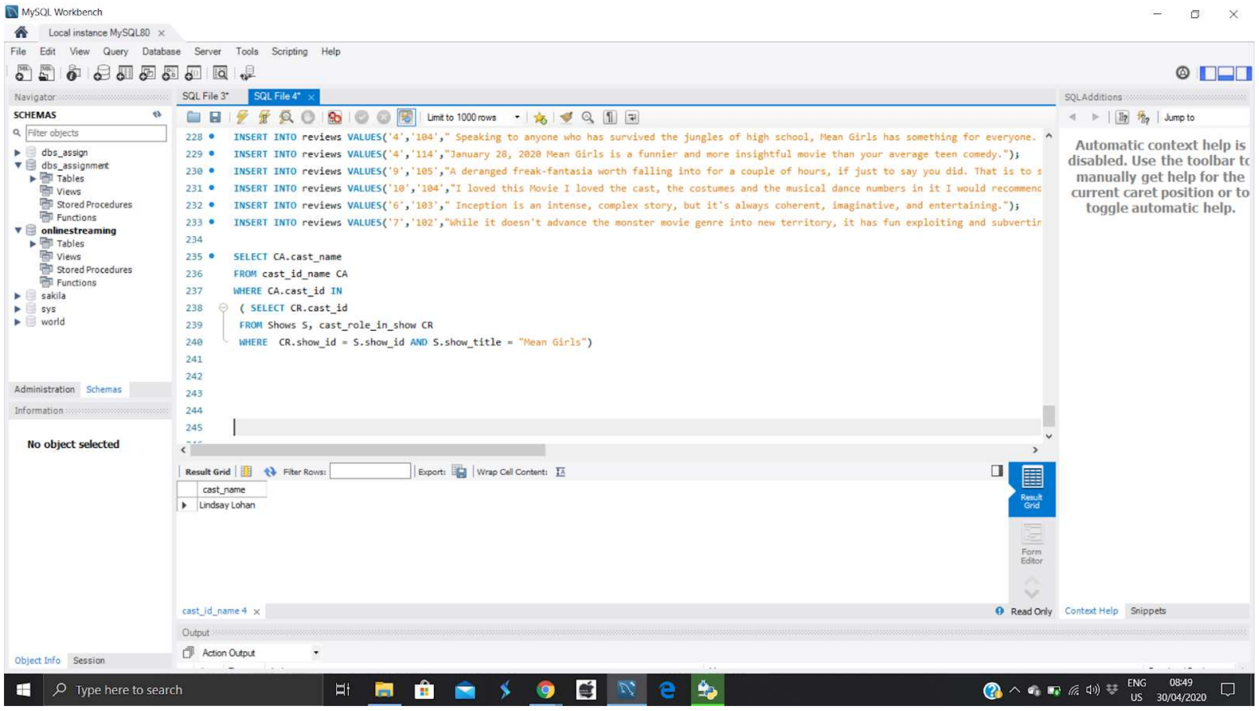
```
select * from subscription;
select * from payment;
select * from cast_role_in_show;
select * from cast_id_name;

select p1.user_id, u1.user_name
from payment p1, user_info u1
where p1.method_of_payment='visa' and p1.user_id=u1.user_id and exists(
select p2.user_ID
from payment p2, subscription s1
where s1.sub_type='ultra' and p2.pay_ID=s1.sub_ID and s1.user_id=u1.user_id);

SELECT CA.cast_name
FROM cast_id_name CA
```

user_id	user_name
u106	ITZY
u114	ATEEZ

```
#    Time      Action                                                                                        Message                                                  Duration / Fetch
224  20:17:57  insert into Devices_registered values('D16', 'Samsung Galaxy Note8', '00:00', '100')          Error Code: 1406. Data too long for column 'checkk' at row 2   0.000 sec
225  20:18:33  select p1.user_id, u1.user_name from payment p1, user_info u1 where p1.method_of_payment='visa' and p1.u...   2 row(s) returned   0.000 sec / 0.000 sec
```

Q2. Write a **correlated query** to display the names of the actors from the film “Mean Girls”.

SELECT CA.cast_name
FROM cast_id_name CA
WHERE CA.cast_id IN
(SELECT CR.cast_id
FROM Shows S, cast_role_in_show CR
WHERE CR.show_id = S.show_id AND S.show_title = "Mean Girls");

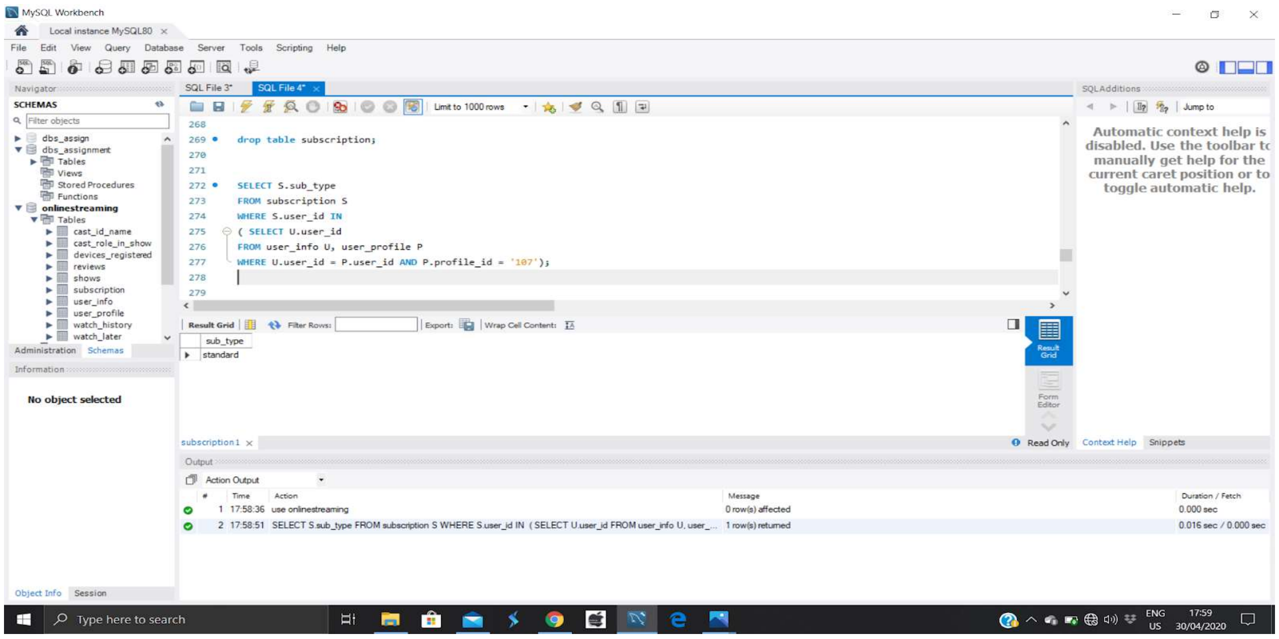


```
INSERT INTO reviews VALUES('4','104'," Speaking to anyone who has survived the jungles of high school, Mean Girls has something for everyone.
INSERT INTO reviews VALUES('4','114',"January 28, 2020 Mean Girls is a funnier and more insightful movie than your average teen comedy.");
INSERT INTO reviews VALUES('9','105',"A deranged freak-fantasia worth falling into for a couple of hours, if just to say you did. That is to s
INSERT INTO reviews VALUES('10','104',"I loved this Movie I loved the cast, the costumes and the musical dance numbers in it I would recommenc
INSERT INTO reviews VALUES('6','103'," Inception is an intense, complex story, but it's always coherent, imaginative, and entertaining.");
INSERT INTO reviews VALUES('7','102',"While it doesn't advance the monster movie genre into new territory, it has fun exploiting and subvertir

SELECT CA.cast_name
FROM cast_id_name CA
WHERE CA.cast_id IN
 ( SELECT CR.cast_id
 FROM Shows S, cast_role_in_show CR
 WHERE  CR.show_id = S.show_id AND S.show_title = "Mean Girls")
```

```
cast_name
Lindsay Lohan
```

Q3. Write a **nested query** to display the subscription type of the user with the profile ID 107.

SELECT S.sub_type
FROM subscription S
WHERE S.user_id IN
(SELECT U.user_id
FROM user_info U, user_profile P
WHERE U.user_id = P.user_id AND P.profile_id = '107');



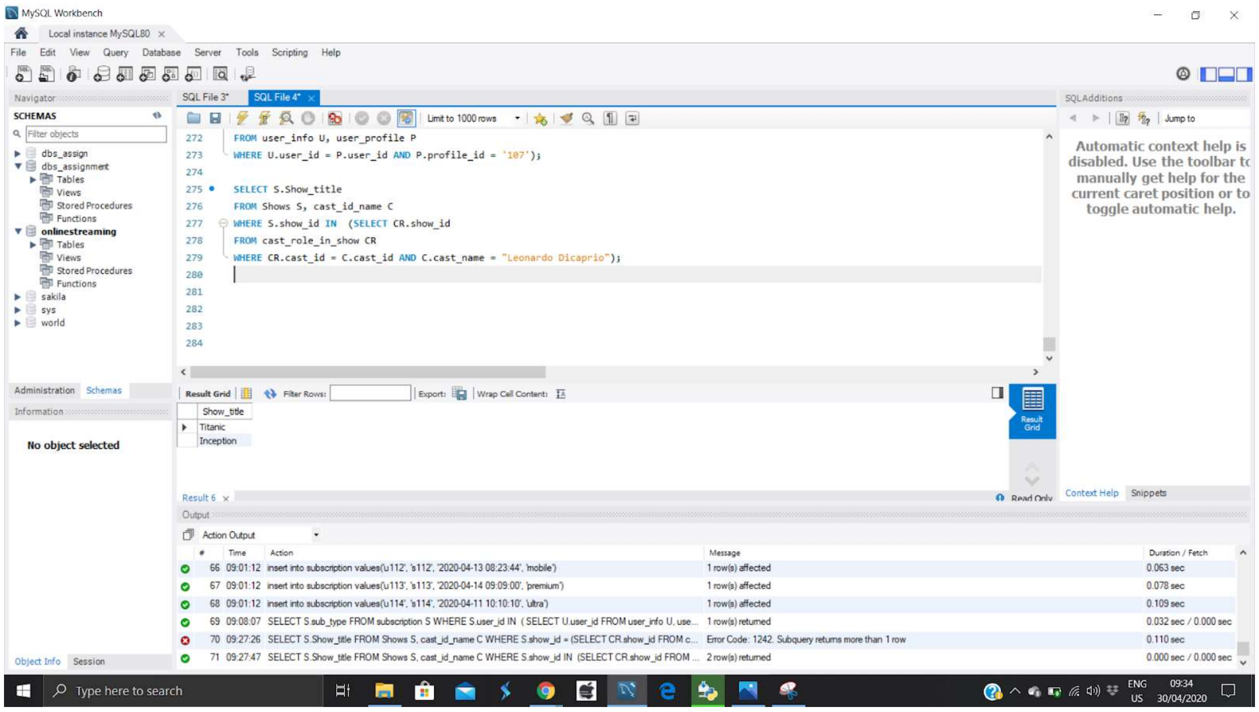
```
drop table subscription;

SELECT S.sub_type
FROM subscription S
WHERE S.user_id IN
( SELECT U.user_id
FROM user_info U, user_profile P
WHERE U.user_id = P.user_id AND P.profile_id = '107');
```

```
sub_type
standard
```

```
#  Time      Action                                                                     Message              Duration / Fetch
1  17:58:36  use onlinestreaming                                                        0 row(s) affected    0.000 sec
2  17:58:51  SELECT S.sub_type FROM subscription S WHERE S.user_id IN ( SELECT U.user_id FROM user_info U, user_...  1 row(s) returned    0.016 sec / 0.000 sec
```

Q4. Write a **correlated query** to display the list of movies that were starred by "Leonardo Dicaprio".

SELECT S.Show_title
FROM Shows S, cast_id_name C
WHERE S.show_id IN (SELECT CR.show_id
FROM cast_role_in_show CR
WHERE CR.cast_id = C.cast_id AND C.cast_name = "Leonardo Dicaprio");



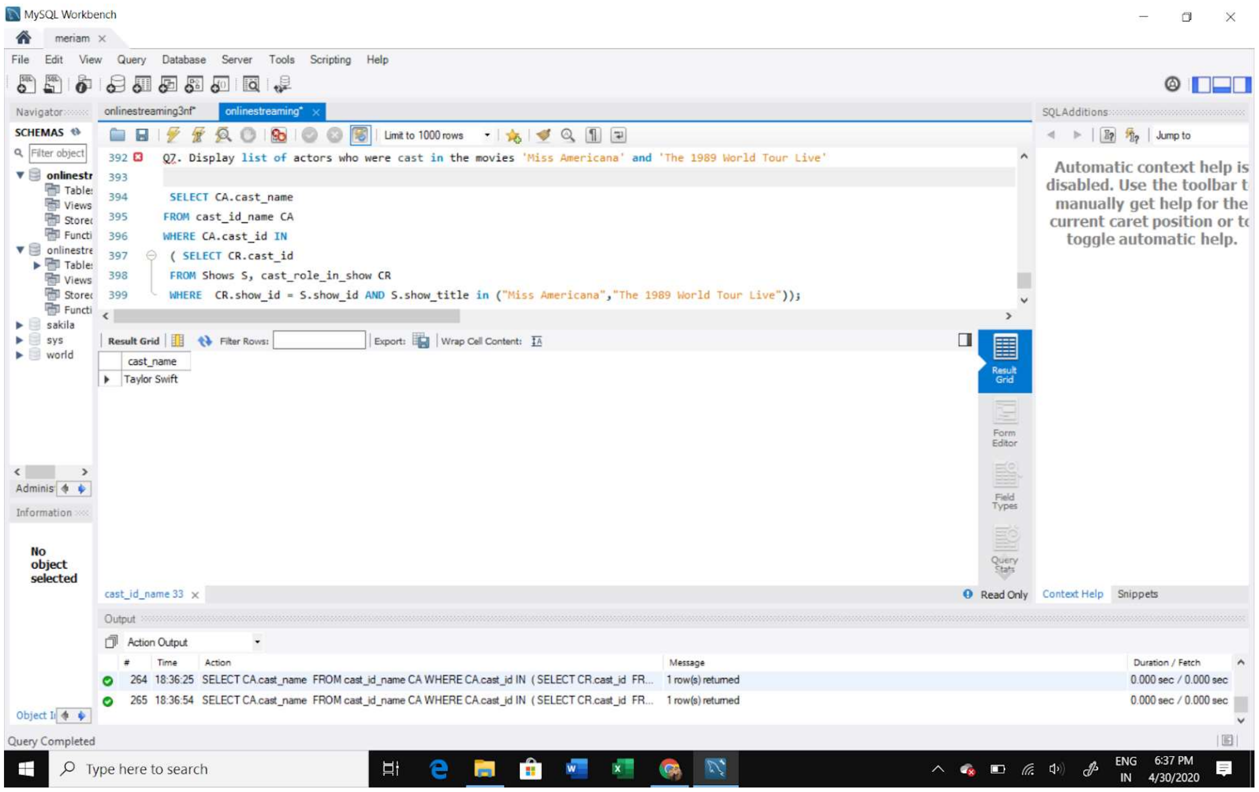
```
FROM user_info U, user_profile P
WHERE U.user_id = P.user_id AND P.profile_id = '107');

SELECT S.Show_title
FROM Shows S, cast_id_name C
WHERE S.show_id IN  (SELECT CR.show_id
FROM cast_role_in_show CR
WHERE CR.cast_id = C.cast_id AND C.cast_name = "Leonardo Dicaprio");
```

```
Show_title
Titanic
Inception
```

```
#   Time      Action                                                                                      Message                                         Duration / Fetch
66  09:01:12  insert into subscription values('u112', 's112', '2020-04-13 08:23:44', 'mobile')              1 row(s) affected                               0.063 sec
67  09:01:12  insert into subscription values('u113', 's113', '2020-04-14 09:09:00', 'premium')             1 row(s) affected                               0.078 sec
68  09:01:12  insert into subscription values('u114', 's114', '2020-04-11 10:10:10', 'ultra')               1 row(s) affected                               0.109 sec
69  09:08:07  SELECT S.sub_type FROM subscription S WHERE S.user_id IN  ( SELECT U.user_id FROM user_info U, use...   1 row(s) returned                  0.032 sec / 0.000 sec
70  09:27:26  SELECT S.Show_title FROM Shows S, cast_id_name C WHERE S.show_id = (SELECT CR.show_id FROM c...   Error Code: 1242. Subquery returns more than 1 row   0.110 sec
71  09:27:47  SELECT S.Show_title FROM Shows S, cast_id_name C WHERE S.show_id IN  (SELECT CR.show_id FROM ...  2 row(s) returned                  0.000 sec / 0.000 sec
```

Q5. Write a **nested query** to display list of actors who were cast in the movies 'Miss Americana' and 'The 1989 World Tour Live'

SELECT CA.cast_name
FROM cast_id_name CA
WHERE CA.cast_id IN
 (SELECT CR.cast_id
 FROM Shows S, cast_role_in_show CR
 WHERE CR.show_id = S.show_id AND S.show_title in ("Miss Americana","The 1989 World Tour Live"));



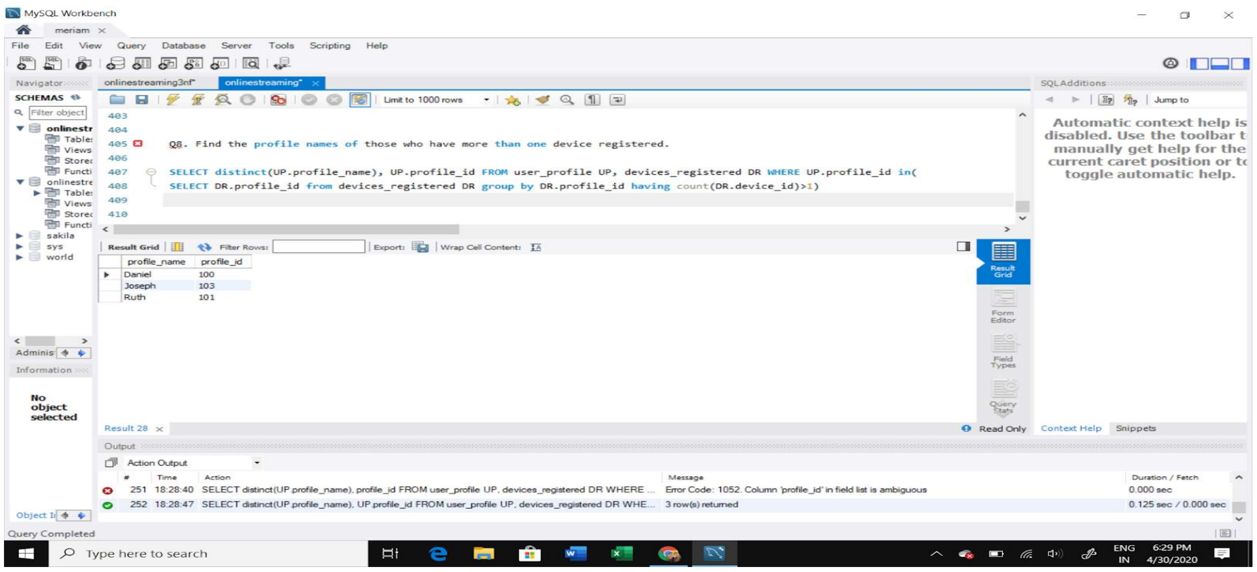
```
Q7. Display list of actors who were cast in the movies 'Miss Americana' and 'The 1989 World Tour Live'

 SELECT CA.cast_name
FROM cast_id_name CA
WHERE CA.cast_id IN
 ( SELECT CR.cast_id
 FROM Shows S, cast_role_in_show CR
 WHERE  CR.show_id = S.show_id AND S.show_title in ("Miss Americana","The 1989 World Tour Live"));
```

cast_name
Taylor Swift

```
#    Time      Action                                                                      Message              Duration / Fetch
264  18:36:25  SELECT CA.cast_name FROM cast_id_name CA WHERE CA.cast_id IN ( SELECT CR.cast_id FR...  1 row(s) returned  0.000 sec / 0.000 sec
265  18:36:54  SELECT CA.cast_name FROM cast_id_name CA WHERE CA.cast_id IN ( SELECT CR.cast_id FR...  1 row(s) returned  0.000 sec / 0.000 sec
```

Q6. Find the profile names of those who have more than one device registered.

SELECT distinct(UP.profile_name), UP.profile_id FROM user_profile UP, devices_registered DR WHERE UP.profile_id in(
SELECT DR.profile_id from devices_registered DR group by DR.profile_id having count(DR.device_id)>1)



```
Q8. Find the profile names of those who have more than one device registered.

SELECT distinct(UP.profile_name), UP.profile_id FROM user_profile UP, devices_registered DR WHERE UP.profile_id in(
SELECT DR.profile_id from devices_registered DR group by DR.profile_id having count(DR.device_id)>1)
```

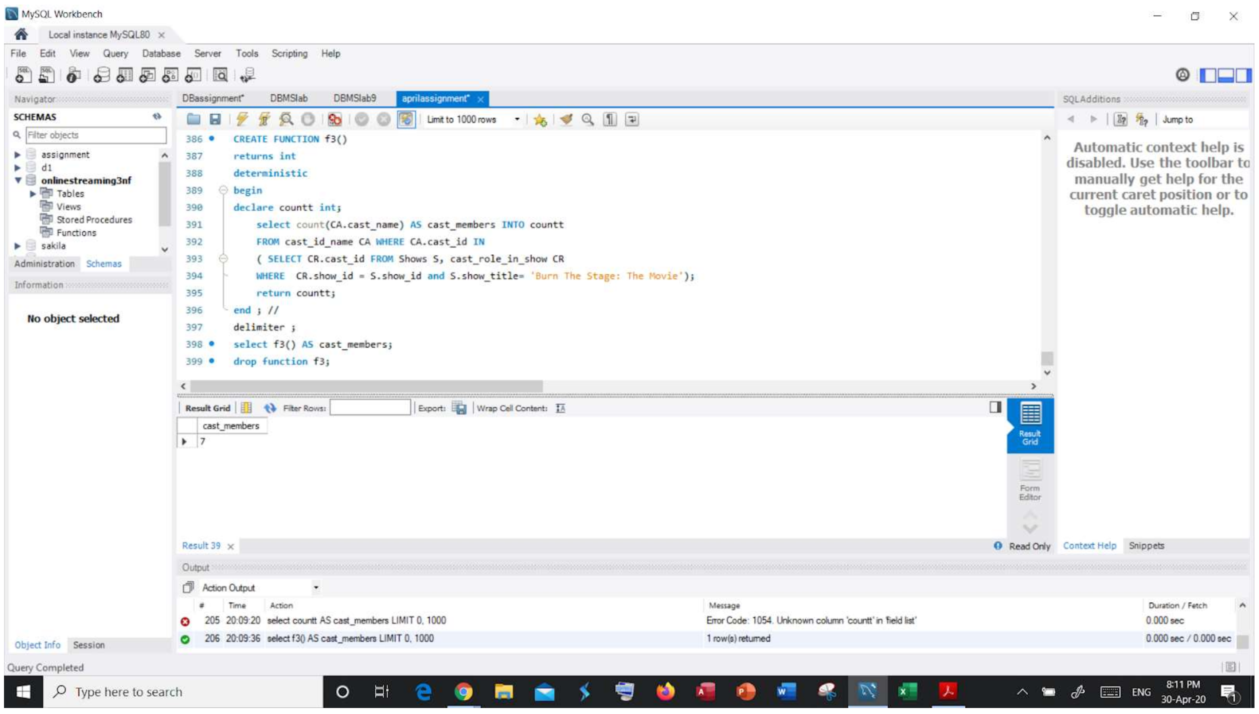
profile_name	profile_id
Daniel	100
Joseph	103
Ruth	101

```
#    Time      Action                                                                                      Message                                                          Duration / Fetch
251  18:28:40  SELECT distinct(UP.profile_name), profile_id FROM user_profile UP, devices_registered DR WHERE ...  Error Code: 1052. Column 'profile_id' in field list is ambiguous  0.000 sec
252  18:28:47  SELECT distinct(UP.profile_name), UP.profile_id FROM user_profile UP, devices_registered DR WHE...  3 row(s) returned                                                0.125 sec / 0.000 sec
```

FUNCTIONS:

Q7. CREATE a **function** to return the number of cast in a show.

```
delimiter //
CREATE FUNCTION f3()
returns int
deterministic
begin
declare countt int;
        select count(CA.cast_name) AS cast_members INTO countt
        FROM cast_id_name CA WHERE CA.cast_id IN
   ( SELECT CR.cast_id FROM Shows S, cast_role_in_show CR
   WHERE  CR.show_id = S.show_id and S.show_title= 'Burn The Stage: The Movie');
        return countt;
end ; //
delimiter ;

select f3() AS cast_members;
```



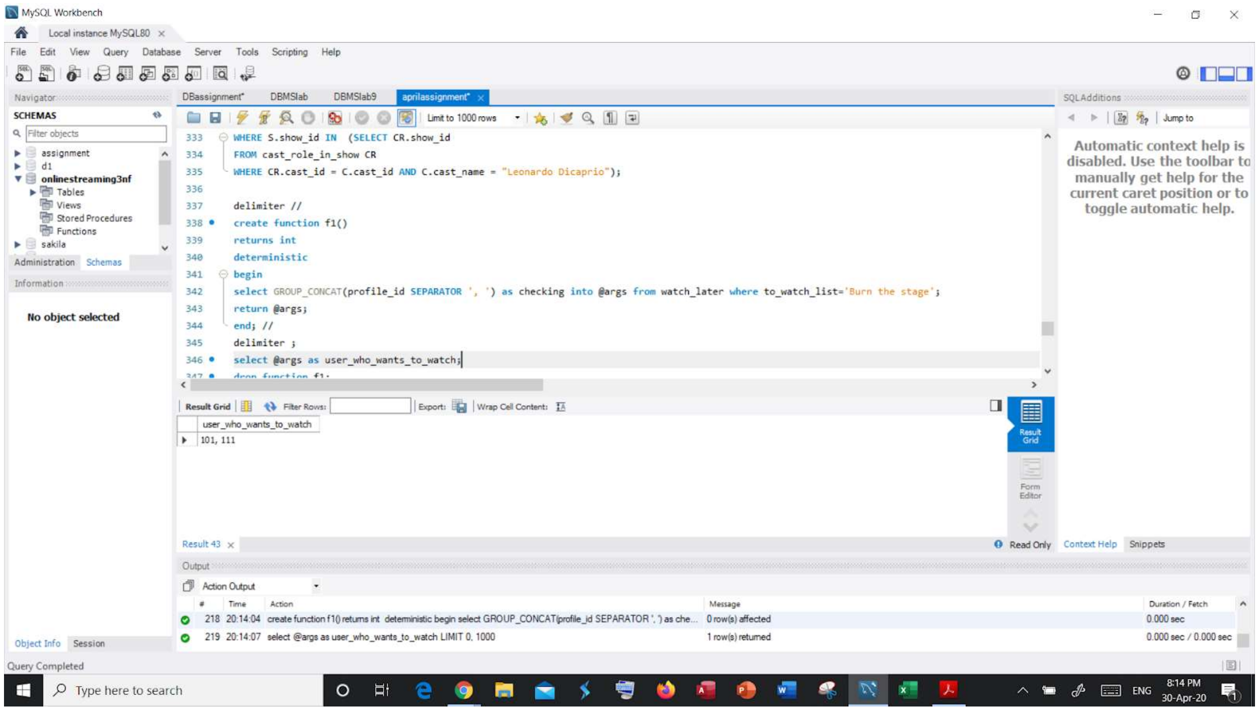
```
CREATE FUNCTION f3()
returns int
deterministic
begin
declare countt int;
    select count(CA.cast_name) AS cast_members INTO countt
    FROM cast_id_name CA WHERE CA.cast_id IN
    ( SELECT CR.cast_id FROM Shows S, cast_role_in_show CR
    WHERE  CR.show_id = S.show_id and S.show_title= 'Burn The Stage: The Movie');
    return countt;
end ; //
delimiter ;
select f3() AS cast_members;
drop function f3;
```

cast_members
7

#	Time	Action	Message	Duration / Fetch
205	20:09:20	select countt AS cast_members LIMIT 0, 1000	Error Code: 1054. Unknown column 'countt' in 'field list'	0.000 sec
206	20:09:36	select f3() AS cast_members LIMIT 0, 1000	1 row(s) returned	0.000 sec / 0.000 sec

Q8. Create a **function** to display the profile IDs of all the people who have 'Burn the stage' in their watch later list.

```
DELIMITER //
CREATE function f1()
RETURNS INT
DETERMINISTIC
BEGIN
SELECT GROUP_CONCAT(profile_id SEPARATOR ', ') AS checking INTO @args
FROM watch_later
WHERE to_watch_list='Burn the stage';
RETURN @args;
END; //
DELIMITER ;

SELECT @args AS user_who_wants_to_watch;
```



Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

```
WHERE S.show_id IN  (SELECT CR.show_id
FROM cast_role_in_show CR
WHERE CR.cast_id = C.cast_id AND C.cast_name = "Leonardo Dicaprio");

delimiter //
create function f1()
returns int
deterministic
begin
select GROUP_CONCAT(profile_id SEPARATOR ', ') as checking into @args from watch_later where to_watch_list='Burn the stage';
return @args;
end; //
delimiter ;
select @args as user_who_wants_to_watch;
drop function f1;
```

user_who_wants_to_watch
101, 111

Output

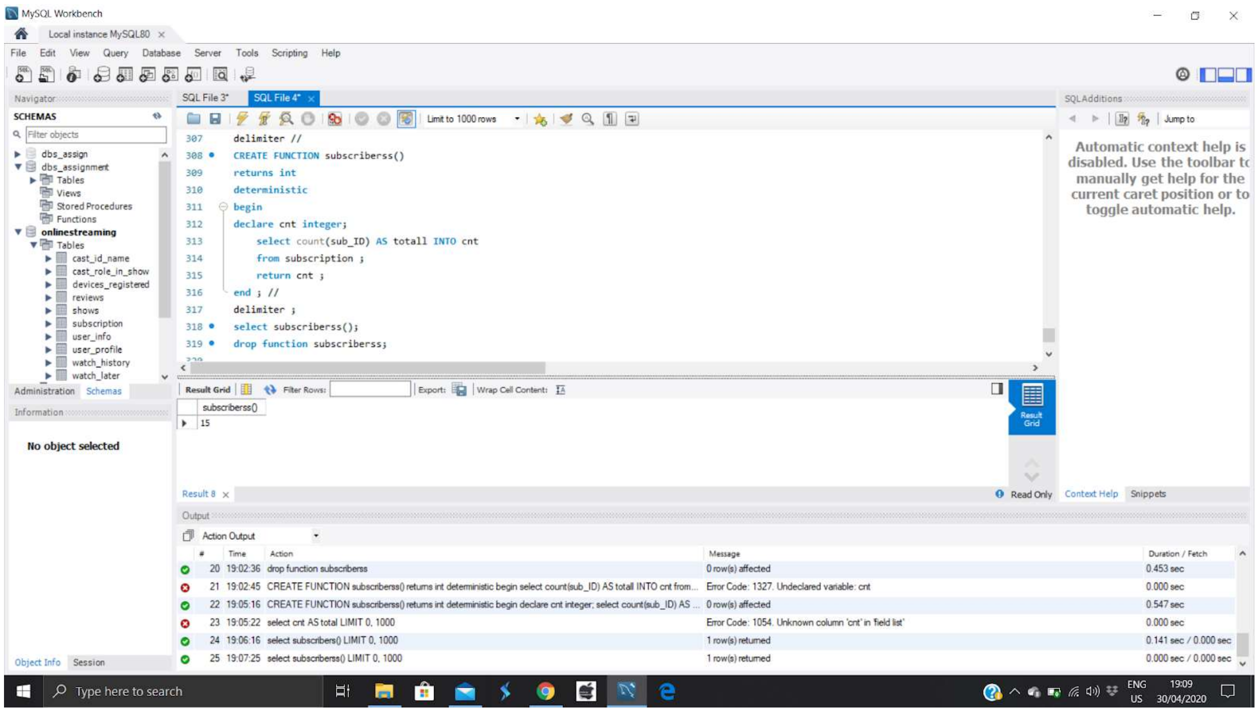
Action Output

#	Time	Action	Message	Duration / Fetch
218	20:14:04	create function f1() returns int deterministic begin select GROUP_CONCAT(profile_id SEPARATOR ', ') as che...	0 row(s) affected	0.000 sec
219	20:14:07	select @args as user_who_wants_to_watch LIMIT 0, 1000	1 row(s) returned	0.000 sec / 0.000 sec

Q9. Create a **function** to display the number of subscribers.

```
CREATE FUNCTION subscriberss()
returns int
deterministic
begin
declare cnt integer;
        select count(sub_ID) AS totall INTO cnt
        from subscription ;
        return cnt ;
```

end ; //
delimiter ;
select subscriberss();



```
delimiter //
CREATE FUNCTION subscriberss()
returns int
deterministic
begin
declare cnt integer;
    select count(sub_ID) AS totall INTO cnt
    from subscription ;
    return cnt ;
end ; //
delimiter ;
select subscriberss();
drop function subscriberss;
```

```
subscriberss()
15
```

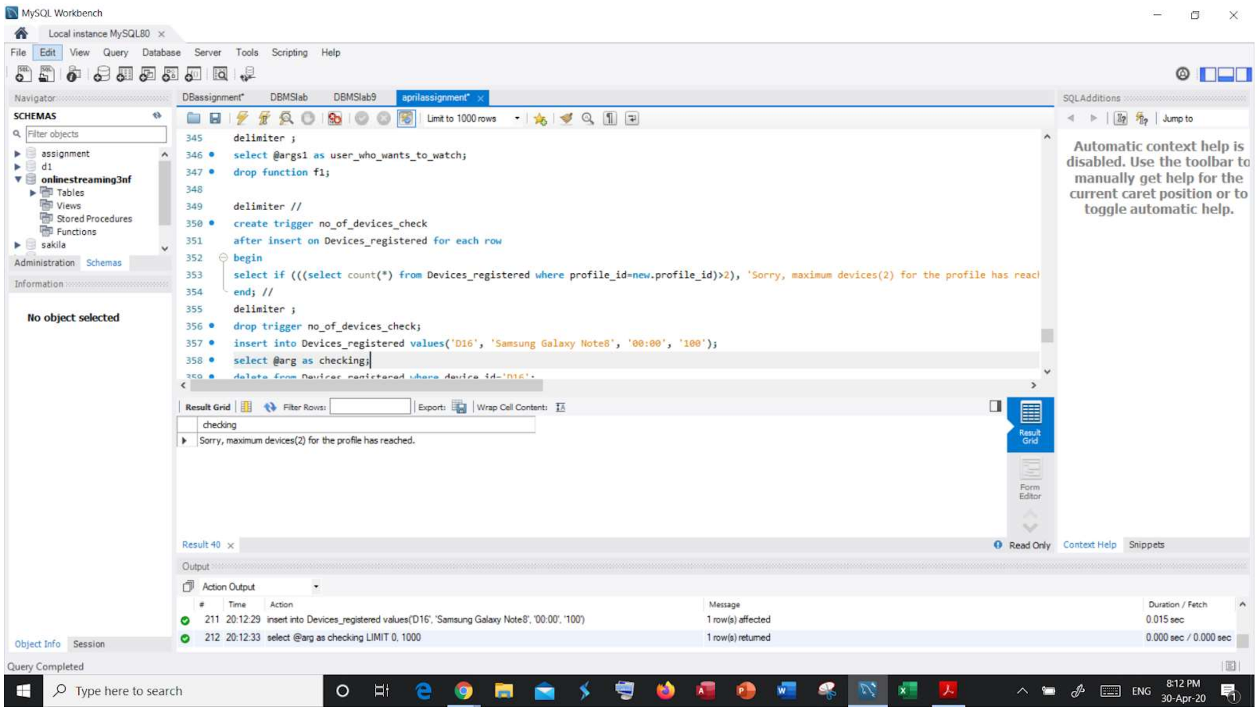
```
#   Time      Action                                                                                          Message                                          Duration / Fetch
20  19:02:36  drop function subscriberss                                                                      0 row(s) affected                                0.453 sec
21  19:02:45  CREATE FUNCTION subscriberss() returns int deterministic begin select count(sub_ID) AS totall INTO cnt from...   Error Code: 1327. Undeclared variable: cnt   0.000 sec
22  19:05:16  CREATE FUNCTION subscriberss() returns int deterministic begin declare cnt integer; select count(sub_ID) AS ...   0 row(s) affected      0.547 sec
23  19:05:22  select cnt AS total LIMIT 0, 1000                                                               Error Code: 1054. Unknown column 'cnt' in 'field list'   0.000 sec
24  19:06:16  select subscribers() LIMIT 0, 1000                                                              1 row(s) returned                                0.141 sec / 0.000 sec
25  19:07:25  select subscriberss() LIMIT 0, 1000                                                             1 row(s) returned                                0.000 sec / 0.000 sec
```

<u>TRIGGER:</u>

Q10. Make a **trigger** to check if the maximum no.of devices have been registered for a profile after insertion.

DELIMITER //
CREATE trigger no_of_devices_check
AFTER INSERT ON Devices_registered FOR EACH ROW
BEGIN
SELECT IF (((
SELECT count(*)
FROM Devices_registered
WHERE profile_id=new.profile_id)>2), 'Sorry, maximum devices(2) for the profile has reached.', 'New device registered!') AS checking INTO @arg;
END; //
DELIMITER ;

INSERT INTO Devices_registered VALUES('D16', 'Samsung Galaxy Note8', '00:00', '100');
SELECT @arg AS checking;



```
delimiter ;
select @args1 as user_who_wants_to_watch;
drop function f1;

delimiter //
create trigger no_of_devices_check
after insert on Devices_registered for each row
begin
select if (((select count(*) from Devices_registered where profile_id=new.profile_id)>2), 'Sorry, maximum devices(2) for the profile has reacl
end; //
delimiter ;
drop trigger no_of_devices_check;
insert into Devices_registered values('D16', 'Samsung Galaxy Note8', '00:00', '100');
select @arg as checking;
delete from Devices_registered where device_id='D16';
```

```
checking
Sorry, maximum devices(2) for the profile has reached.
```

```
#    Time      Action                                                                        Message             Duration / Fetch
211  20:12:29  insert into Devices_registered values('D16', 'Samsung Galaxy Note8', '00:00', '100')  1 row(s) affected   0.015 sec
212  20:12:33  select @arg as checking LIMIT 0, 1000                                          1 row(s) returned   0.000 sec / 0.000 sec
```